sustainability

MDPI

Article

# $PM_{2.5}$ Concentration Prediction in the Cities of China Using Multi-Scale Feature Learning Networks and Transformer Framework

**Zhaohan Wang [1,2], Kai Jia [1,3,*], Wenpeng Zhang [1] and Chen Zhang [3]**

[1] School of Artificial Intelligence and Software Engineering, Nanyang Normal University, Nanyang 473000, China; 20131044@nynu.edu.cn (W.Z.)

[2] School of Computer Science and Engineering, University of New South Wales, Sydney, NSW 2052, Australia; zhaohan.wang1@student.unsw.edu.au (Z.W.)

[3] Hubei Key Laboratory of Transportation of Internet of Things, School of Computer Science and Artificial Intelligence, Wuhan University of Technology, Wuhan 430070, China; zhangchenorange@whut.edu.cn

* Correspondence: jiakai@nynu.edu.cn

## Abstract

Particulate matter (PM) concentration, especially $PM_{2.5}$, is a major culprit of environmental pollution from unreasonable energy system emissions that significantly affects visibility, climate, and public health. The prediction of $PM_{2.5}$ concentration holds significant importance in the early warning and management of severe air pollution, since it enables the provision of guidance for scientific decision-making through the estimation of impending $PM_{2.5}$ concentration. However, due to diversified human activities, seasonal factors and industrial emissions, the air quality data not only show local anomalous mutability, but also global dynamic change characteristics. This hinders existing $PM_{2.5}$ prediction models from fully capturing the aforementioned characteristics, thereby deteriorating the model performance. To address these issues, this study proposes a framework integrating multi-scale temporal convolutional networks (TCNs) and a transformer network (called MSTTNet) for $PM_{2.5}$ concentration prediction. Specifically, MSTTNet uses multi-scale TCNs to capture the local correlations of meteorological and pollutant data in a fine-grained manner, while using transformers to capture the global temporal relationships. The proposed MSTTNet's performance has been validated on various air quality benchmark datasets in the cities of China, including Beijing, Shanghai, Chengdu, and Guangzhou, by comparing to its eight compared models. Comprehensive experiments confirm that the MSTTNet model can improve the prediction performance of 2.42%, 2.17%, 2.87%, and 0.34%, respectively, with respect to four evaluation indicators (i.e., Mean Absolute Error, Root Mean Square Error, Mean Absolute Percentage Error, and R-square), relative to the optimal baseline model. These results confirm MSTTNet's effectiveness in improving the accuracy of $PM_{2.5}$ concentration prediction.

**Keywords:** air pollution; $PM_{2.5}$ concentration prediction; temporal convolution network; transformer

check for updates

Academic Editor: Jan Hopmans

Received: 21 August 2025
Revised: 2 October 2025
Accepted: 4 October 2025
Published: 6 October 2025

**Citation:** Wang, Z.; Jia, K.; Zhang, W.; Zhang, C. $PM_{2.5}$ Concentration Prediction in the Cities of China Using Multi-Scale Feature Learning Networks and Transformer Framework. *Sustainability* **2025**, *17*, 8891. https://doi.org/10.3390/su17198891

## 1. Introduction

Air pollution, especially fine particulate matter (e.g., $PM_{2.5}$), has become a serious global environmental problem due to its severe impacts on public health, climate, and visibility [1]. $PM_{2.5}$, characterized as particulate matter with an aerodynamic diameter of 2.5 micrometers or less, stands as a pivotal etiological factor in respiratory and cardiovascular diseases [2], e.g., malignant neoplasm of trachea, bronchus, and lung, and is also closely related to premature mortality, e.g., other ischemic heart diseases, pulmonary embolism, and cerebrovascular diseases [2]. Accurate prediction of $PM_{2.5}$ concentrations is indispensable to effective air quality governance, facilitating the implementation of timely intervention strategies and pollution mitigation measures. Moreover, efficient and effective PM prediction can strongly complement the advantages of these satellite monitoring platforms, e.g., TEMPO, CINDI-3, Sentinel-4, and GOME. These platforms can monitor key gaseous precursors of PM (e.g., $NO_2$ and $SO_2$) and aerosol optical depth (AOD) at high frequency and high resolution, and the predicted PM data can be used to verify and calibrate PM estimation products derived from satellite AOD data. However, the dynamic and non-linear characteristics of $PM_{2.5}$ are affected by the complex interactions among meteorological conditions, industrial emissions, and socioeconomic activities, posing significant challenges to traditional prediction models.

In retrospect, many methods have been developed for the task of $PM_{2.5}$ concentration prediction, such as the autoregressive integrated moving average (ARIMA) model [3], multi-layer perceptron (MLP) [4], convolutional neural network (CNN) [5,6], recurrent neural network (RNN) [7,8] and its variants (e.g., long short-term memory (LSTM) [9,10], gate recurrent unit (GRU) [11,12], and among others. However, the task of $PM_{2.5}$ prediction is troublesome since air quality time series data change dynamically with various external factors, such as human activities, seasonal factors, and industrial emissions.

Classical statistical analysis methods, including linear regression [13] and ARIMA [3], are easy to comprehend and implement, and have been extensively employed in the scenarios of air quality prediction. However, these models generally rest on the assumption that a linear correlation exists among the variations in air quality data. They struggle to process the dynamic and high-dimensional data, resulting in degraded model performance and adaptability. Ultimately, according to the constraints of linear functions, local non-linear characteristics and long-term temporal dependencies may fail to be completely captured [14,15]. Recently, some classical machine learning methods, such as support vector regression (SVR) [16,17] and MLP [4], have also been widely employed in air quality prediction tasks. In this regard, SVR and MLP enable modeling the non-linear transformation relationships of input and output during training stage. The prediction ability is greatly improved compared to statistical analysis methods. Nevertheless, these models tend to encounter gradient-related issues and fall into local optimal solutions during the training phase. This may potentially lead to the deterioration of prediction accuracy for such machine learning models in air quality prediction tasks. Therefore, for the first two classes of methods, they are easy to implement, but their performance is limited because they cannot achieve deep feature extraction for air quality prediction. Moreover, they cannot capture the dynamic changes among indicators, which also degrades model performance.

With the advancements in artificial intelligence (AI) and deep learning, data-driven techniques have risen to prominence owing to their capacity to capture non-linear relationships and temporal dependencies. In the context of our research on $PM_{2.5}$ concentrations prediction, the importance of using an AI-based model stems from its ability to tackle problems that are intractable for traditional physical/statistical models. Specifically, AI-based models can process high-dimensional, non-linear feature data [18–21]. Air pollution systems are influenced by a multitude of interacting factors (e.g., emissions, meteorology, chemistry, topography). The relationships among these factors are highly non-linear and high-dimensional. Traditional linear models (e.g., linear regression [13] and ARIMA [3]) often fail to capture these intricate patterns. AI-based models excel at automatically extracting key features from data without requiring prior physical assumptions, thereby achieving effective predictions. Models such as LSTM [10], GRU [11], CNN [5], bidirectional LSTM (BiLSTM) [18], and their hybrid variants [19–21] have been proven to have superior performance in $PM_{2.5}$ prediction by leveraging their capacity to process complex air quality data. Within these models, CNNs are employed to model local or spatial correlations within air quality time series [20]. However, CNN-equipped models often exhibit a limited capacity to capture temporal and long-term dependencies, attributable to the restricted receptive field using fixed convolution kernel and single scale form [22]. The RNN model and its variants are employed to capture temporal dependencies within air quality data [10,11,18], but lack global modeling capabilities [23,24]. In practical meteorological monitoring process, most air quality data, e.g., humidity, wind speed, and pressure conditions, show local anomalous mutability and global dynamic changes in different monitoring stations [25]. These data features signify unprecedented changes impacted by heterogeneous external factors over time. This may lead to the omission of relevant and critical information during the training process, and some noisy data will also affect model learning, thereby deteriorating model performance. In summary, the model that has fine-grained extraction abilities should be constructed to capture both local anomalous mutability and global dynamic variation features, achieving high-precision prediction.

To address the aforementioned issues in air quality prediction tasks, this study presents a novel framework, called MSTTNet, which innovatively integrates multi-scale temporal convolutional networks (TCNs) and transformer components, which can capture local irregular variability and global dynamic variations in meteorological data. On the one hand, the TCN component is capable of reducing the constraint of fixed receptive fields [22], and capturing the intrinsic local characteristic in meteorology data. Currently, benefiting from the architectural advantages, TCNs are being adopted in sequence modeling tasks, e.g., sequence prediction and classification [26–30], natural language processing (NLP) [31], and medical image processing [32]. Transformers can model global correlations and weaken the impact of noisy data on the model's learning ability, which is broadly applied in computer vision [33], pattern recognition [34], and sequence modeling [35–39]. In this paper, the self-attention mechanism within the transformer block is able to process arbitrary portions of air quality data without distance constraints, thereby endowing it with a stronger capability to capture global dependencies.

Specifically, MSTTNet first employs two TCN blocks to capture local features from the original air quality data. In each TCN block, a multi-scale architecture is designed to extract local and spatial feature information of different scales in a fine-grained manner by adopting different filter sizes, and the extracted representational information from different scales in two TCN blocks is then fused. Subsequently, the fused features are fed into the transformer part to adaptively learn and model the importance of features and time steps, thereby facilitating the attenuation of the impact of noisy information. Finally, the flatten layer and linear layer with one neuron are used to achieve $PM_{2.5}$ prediction for the next time slice. The performance of MSTTNet is tested among four real-world benchmark air quality datasets, including Beijing, Shanghai, Chengdu, and Guangzhou. The comprehensive experiments demonstrate that the proposed MSTTNet achieves an average improvement of 2.42%, 2.17%, 2.87%, and 0.34%, respectively, across four evaluation indicators, i.e., MAE, RMSE, MAPE, and $R^2$, compared with optimal baseline model BiLSTM. The key contributions for this study are summarized as follows.

- A new air quality prediction framework called MSTTNet is proposed to model the local correlation and global dynamic change characteristics of air quality data in a fine-grained manner.
- MSTTNet innovatively integrates the multi-scale TCNs and transformer based on their architectural advantages. Multi-scale TCNs are designed to extract local anoma-

lous mutability and spatial information for different dimensions. The transformer is adopted to adaptively learn and capture the significance of time steps and global features, enabling the weakening of the impact of noise information.

- Extensive experiments are performed among four benchmark air quality datasets to verify the prediction capacity of the MSTTNet. Numerical analyses manifest that the MSTTNet has superiority in comparison with its eight competitors.

The rest of this paper is organized as follows. In Section 2, the related works on air quality prediction are introduced in detail. Section 3 describes the architecture of proposed MSTTNet. Section 4 reports the experimental details and results of the models, and analyzes the performance difference among models. Finally, Section 5 describes the conclusion and future work.

## 2. Related Work

Currently, air quality prediction is a hot research area, since it can assist government authorities and decision-makers in the comprehensive management of emission control, traffic management, urban planning, and many others. This prediction mode can also remind residents to reduce $PM_{2.5}$ exposure, and protect public health. By reviewing existing models in air quality prediction, they can be categorized into three classes: statistical analysis-based models, machine learning-based models, and hybrid models.

### *2.1. Statistical Analysis Models*

This type of model, such as ARIMA, usually has good interpretability and high computational efficiency, and is often used to predict small amounts of data in the early years. For instance, Gourav et al. [40] employed the ARIMA time series technique to model and predict the monthly future air quality in New Delhi, India. In response to the limitations of the ARIMA model, subsequent studies have proposed a variety of improvement solutions. Aladağ [41] combined wavelet transform with ARIMA to perform air quality prediction, which achieved better results than traditional ARIMA. The study first used wavelet decomposition to decompose the original air quality series into components of different frequencies, then established ARIMA models for each component, and finally reconstructed the prediction results [41]. Cekim [3] employed various time series models to predict $PM_{10}$ concentrations in 2019 in Hatay and Yalova, and confirmed that singular spectrum analysis (SSA) is the optimal model.

Furthermore, several linear regression methods have also been employed for the task of air quality prediction. For example, Abdullah et al. [42] used a multiple linear regression (MLR) model to predict $PM_{10}$ concentrations in the transboundary haze events, and confirmed that the MLR model achieved the best prediction performance with lower fitting errors. In addition to the above methods, other statistical techniques are often used in air quality prediction. Zhou et al. [43] designed a seasonal grey model to predict air quality time series in the Yangtze River Delta. Comparative experiments demonstrated that the designed model is superior to other competing models in enhancing the prediction performance when dealing with seasonal air quality variations.

However, air quality data exhibit highly non-linear and complex spatiotemporal dependencies. This will cause the prediction accuracy of traditional statistical methods to drop significantly, prompting researchers to turn to more advanced machine learning models.

### *2.2. Machine Learning Models*

The revolution of computing power and big data technology has allowed machine learning models, e.g., MLP [4] and RNN [10], to perform exceptionally well in air quality prediction. The superior fitting ability of these machine learning models enables them to possess the accuracy that is difficult to achieve by traditional statistical analysis models when dealing with non-linear and non-stationary air quality time series.

For instance, Talepour et al. [44] verified the prediction accuracy of MLR and MLP models in predicting $PM_{10}$ and $PM_{2.5}$ levels. Experiment results showcase that the MLP has impressive performance compared to the MLR model. The gradient issues inherent in RNN models have been relieved by the adoption of sophisticated variants, e.g., LSTM, GRU, and BiLSTM [10]. He et al. [10] implemented a suite of machine learning models, encompassing LSTM, GRU, BiLSTM, and CNN–LSTM hybrid models, to predict the indoor $PM_{2.5}$ concentrations in shared office spaces. This study has assessed for robustness, uncertainty, and feature importance, and confirmed the outstanding predictive ability of LSTM over traditional mass balance models and other comparative models. Zheng et al. [45] first applied a full-connection LSTM model in $PM_{2.5}$ concentration prediction for four major cities in China (i.e., Beijing, Shanghai, Guangzhou, and Shenyang) using previous air quality monitoring data. The proposed model is capable of predicting $PM_{2.5}$ concentrations over the subsequent 24-hour period by leveraging 72 hours of historical monitoring data. Moreover, the merits of temporal convolutional networks (TCNs) in sequence processing have been substantiated [22]. Specifically, TCNs not only showcase exceptional competence in temporal feature extraction but also display prominent efficiency in computational performance. Samal et al. [46] developed a TCN with an imputation block (TCN-I), to simultaneously perform data imputation and prediction tasks. The numerical results confirmed that the TCN-I model outperforms the baseline models. However, single models typically exhibit limited generalization capability in parallel data processing and continue to encounter obstacles in deriving meaningful patterns from complex and dynamic air quality monitoring data.

### 2.3. Hybrid Models

The single model is often difficult to comprehensively extract the complex characteristics of air quality data, so hybrid methods that combine the advantages of multiple models have become a current research hotspot. The hybrid model is capable of alleviating the issues of inadequate predictive capability and unstable generalization performance inherent in a single model through the integration of diverse technologies, thereby attaining enhanced prediction accuracy [19–21]. For instance, Pak et al. [47] integrated a spatiotemporal CNN with an LSTM neural network (called CNN-LSTM) to predict the daily average $PM_{2.5}$ concentration in Beijing. The CNN-LSTM predictor exhibits proficiency in extracting the intrinsic features and long-term temporal dependencies from air quality and meteorological input data. Zhu et al. [20] designed an automated hourly $PM_{2.5}$ prediction model by integrating 1DCNN-BiLSTM models utilizing input data from both the target monitoring station and its adjacent sites. Experimental analysis conformed that proposed model can promote the prediction ability. Due to the advantage of graph convolutional network (GCN) to extract features from non-Euclidean spaces, so it is adept at modeling the spatial features of air quality data. Qi et al. [48] adopted a hybrid framework GC-LSTM that combines the GCN network and LSTM network to perform hourly $PM_{2.5}$ concentration prediction. The experimental analysis demonstrated that the GC-LSTM model is capable of enhancing prediction performance. In comparison to conventional CNNs, the TCNs are capable of expanding the receptive field through dilated and causal convolutions, thereby enhancing the effectiveness of feature extraction. For example, Ren et al. [49] adopted a hybrid framework that incorporates the TCN module and LSTM module for predicting PM concentrations in Xi’an City. The TCN module was first employed to extract features from the influence factors of PM, after which the LSTM network was leveraged to learn from the TCN-derived high-level features, thereby enabling the prediction of PM concentrations.

In addition, models based on attention mechanisms are also widely used. Li et al. [50] proposed a $PM_{2.5}$ concentration prediction method based on the RCG-attention model, wherein the residual neural network and convolutional GRU are employed to learn spatial and temporal features, respectively, and finally the multi-dimensional features can be obtained by the attention mechanism.

Different from these works, this study adopts a multi-scale TCNs model with different input channels to extract local anomalous fluctuations and spatial features in a fine-grained manner. The extracted features are then modeled through the transformer with self-attention mechanism to capture long-term dynamic changes. This framework focuses on further promoting the air quality prediction ability while remedying the weaknesses of traditional single models, e.g., CNN and LSTM.

## 3. Proposed Method

This section mainly describes the architecture of the proposed MSTTNet framework to solve the problems mentioned in the air quality prediction tasks. The developed MSTTNet framework effectively establishes predictive correlations between prior air quality metrics and subsequent $PM_{2.5}$ concentration levels at future hourly time intervals. The architectural implementation of the proposed MSTTNet framework, as illustrated in Figure 1, comprises three key components: data preprocessing, feature extraction, and $PM_{2.5}$ concentration prediction.

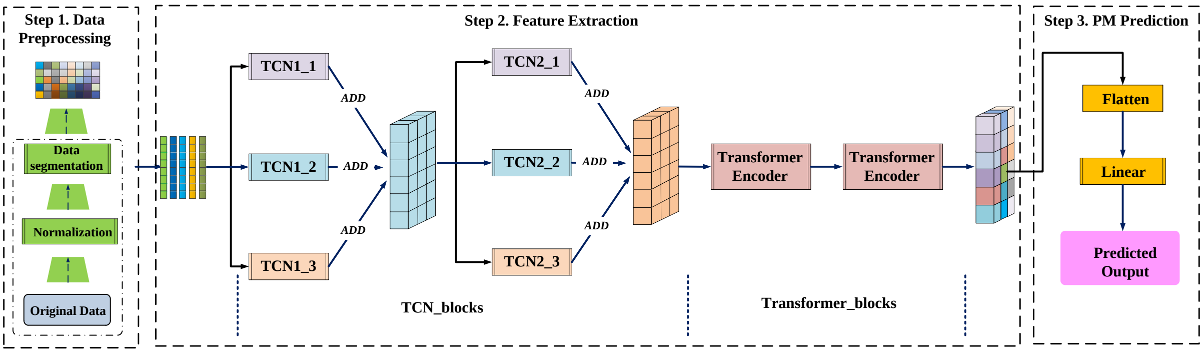


**Figure 1.** The architecture of MSTTNet, including data preprocessing, feature extraction, and PM prediction steps.

Specifically, the raw air quality data undergoes preprocessing prior to being fed into the neural network for model training. This stage incorporates two essential processing techniques: normalization operation and moving window sampling. Then, the multi-scale TCNs module is constructed by integrating various TCN blocks with different filter sizes to capture the local anomalous fluctuations and spatial relationships in a fine-grained manner. Later, the transformer encoder is adopted to adaptively learn and capture the significance of time steps and global features, enabling the weakening of the impact of noise information. The final prediction of $PM_{2.5}$ concentration is generated by processing the fused high-level features through a dedicated prediction module, which sequentially incorporates a flatten layer and a single-neuron fully-connected layer. The concrete construction processes of each step are expounded as below.

### 3.1. Step 1: Data Preprocessing

*Data Normalization*: The original data from different sensors is assembled into a multi-dimensional time series, including variables such as atmospheric pressure, humidity, and PM concentration. Due to the multi-sensor acquisition of monitoring indicators, the collected data typically exhibit significant variations in scale and dimensionality. To mitigate the adverse effects of such heterogeneity on model training performance, data normalization represents an essential preprocessing step for sequence modeling tasks [20], as it standardizes the input features within a uniform numerical range. For notational clarity, let $n$ denote the number of air quality data samples and $m$ represent the dimensionality of the feature space (i.e., the number of monitored variables). Formally, the original air quality data is denoted as $X = [x_1,x_2,x_3,\ldots,x_i,\ldots,x_n]^T$, $x_i = [x_{i,1},x_{i,2},x_{i,3}\ldots,x_{i,j},\ldots,x_{i,m-1},x_{i,m}]$, wherein $x_{i,j}$ indicates the measured value of the $j$-th air quality indicator at time step $i$. Following standard preprocessing practices [9], we apply min-max normalization to scale all features uniformly to the interval [0, 1], as calculated in Equation (1).

$$x_{(i,j)}^{norm} = \frac{x_{(i,j)} - x_{(i,j)}^{min}}{x_{(i,j)}^{max} - x_{(i,j)}^{min}} \tag{1}$$

where $x_{(i,j)}$ and $x_{(i,j)}^{norm}$ are the original and normalized air quality data, $x_{(i,j)}^{max}$ and $x_{(i,j)}^{min}$ denote the maximum and minimum values observed in the air quality feature space, respectively.

*Moving window sampling*: Air quality prediction is essentially a typical sequence prediction task, necessitating the transformation of multivariate time series data into supervised learning samples through appropriate feature engineering. The moving window sampling technique is adopted in this work, since this method can explore the temporal relationship of air quality time series, and can obtain more sample data for model training. Figure 2 illustrates the operational mechanism of the moving window sampling technique employed in this study. It should be noted that the prediction target of each input sample, i.e., features, is the PM concentration at the next moment. $L$ denotes the lookback window size (i.e., time lag) for historical observations, while $m$ corresponds to the feature space dimensionality (i.e., number of measured variables).

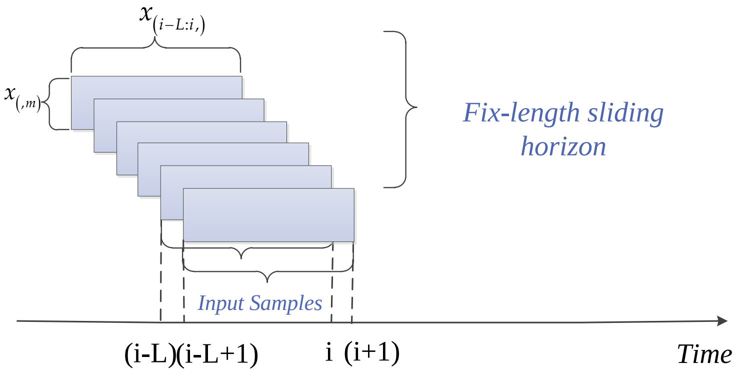


**Figure 2.** Operational mechanism of the moving window sampling technique.

### 3.2. Step 2: Feature Extraction

- *1. Multi-scale TCNs block*

A temporal convolution network (TCN) is an evolutionary structure for the classical CNN architecture. The comparisons between traditional CNN and TCN are depicted in Figure 3. To expand the receptive field in conventional CNN architectures, additional convolutional layers must be sequentially stacked layer-by-layer. Inevitably, the progressive deepening of network layers inherently leads to extreme growth in model parameters, which consequently demands greater computational resources for training while potentially exacerbating gradient-related optimization challenges. In contrast, TCN architecture achieves receptive field expansion through adjustable dilation factors, which is a lightweight design and does not increase the trainable model parameters.

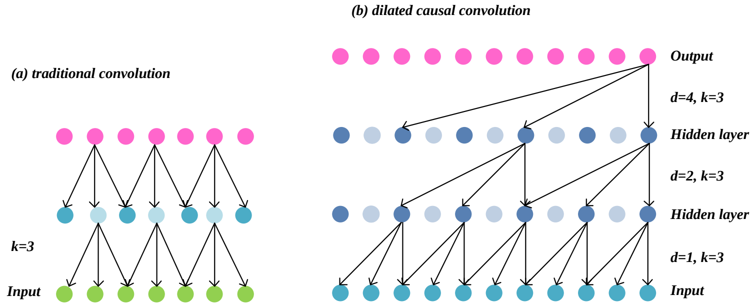


**Figure 3.** Detail description of traditional CNN and TCN structures.

The TCN architecture contains two elaborate components, namely causal convolution and dilated convolution. The former can ensure strict temporal dependency in the prediction process. This design guarantees temporal causality in sequence processing, where the output at time step $t$ is strictly conditioned on historical inputs from preceding time steps, which is a significant distinction from the traditional CNN architecture, as illustrated in Figure 3. This causal modeling approach effectively prevents temporal information leakage, making it particularly suitable for sequential prediction tasks, e.g., time series prediction and speech signal generation. The latter is dilated convolution, which can capture the dependencies between different time steps in a longer range while maintaining constant parameter complexity and computational overhead (i.e., dilated coefficient). Therefore, this architecture enables fine-grained modeling of both local temporal patterns and spatial correlations within air quality time series data. In contrast to conventional CNN architecture, dilated convolution achieves expanded receptive fields through adjustable dilation factors while more effectively capturing temporal dependencies within sequential data. The TCN architecture synergistically integrates these advantages, preserving strict temporal causality while employing dilated convolutions to extract multi-scale features. This combined structure enhances the model's capacity for local–spatial relationship modeling without compromising computational efficiency. Given an input sample $x$, for element $s$ at time step $t$ in sample $x$, the convolution calculation of $DCC$ [22] for a filter $f$ is represented as in Equation (2):

$$F(s) = (x *_d f)(s) = \sum_{i=0}^{ks-1} f(i) \cdot x_{s-d \cdot i} \tag{2}$$

Among them, $*$ indicates the convolution calculation, $d$ represents the dilation factor governing receptive field expansion, $k$ represents the size of filters, and $s - d \cdot i$ illustrates the direction of the historical data. Figure 4 describes the TCN unit structure used in this paper, including the residual mapping branch and identity mapping branch. Formally, the calculation procedure for the $n$-th layer $DCC$ in each TCN block can be represented as in Equation (3):

$$DCC^{(n)} = Conv1D(W^{(n)}, b^{(n)}, input_{x_i}, kernel\_size = ks, dilation\_rate = d) \tag{3}$$

After activation, batch normalization, and dropout layers, the extracted features of each TCN unit can be obtained by adding the output of the residual mapping branch (denoted as $DCC$) and identity mapping path (denoted as $H(x_i)$), as calculated in Equation (4):

$$O^{i}_{tcn\_unit} = DCC + H(input_{x_i}) \tag{4}$$

where $i$ represents the $i$-th TCN unit, $i \in 1, 2, 3$. Since different scales can model local information of different granularities of air quality data, we adopt a multi-scale architecture to perform feature extraction in parallel, as described in Figure 5, in which $ks\_i$ indicates the kernel size of the $i$-th TCN unit, and $fn$ denotes the number of convolution filters. Finally, the output of each multi-scale TCN block can be obtained by fusing the outputs of different TCN units, as calculated in Equation (5):

$$h^{n}_{multiscale-tcn} = Add\left(O^{1}_{tcn\_unit}, O^{2}_{tcn\_unit}, O^{3}_{tcn\_unit}\right) \tag{5}$$

where $n$ represents the $n$-th layer of multi-scale convolutional blocks, and $h^{n}_{multiscale-tcn}$ represents the fused high-level features from different TCN units.

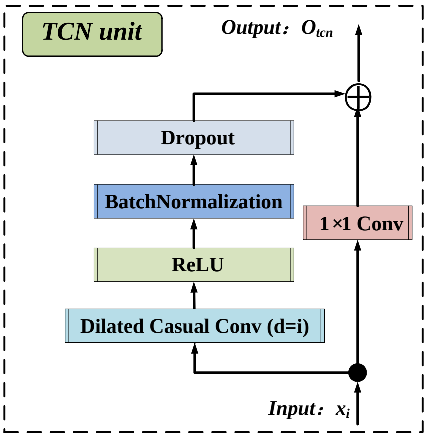


**Figure 4.** An illustration of the TCN unit.

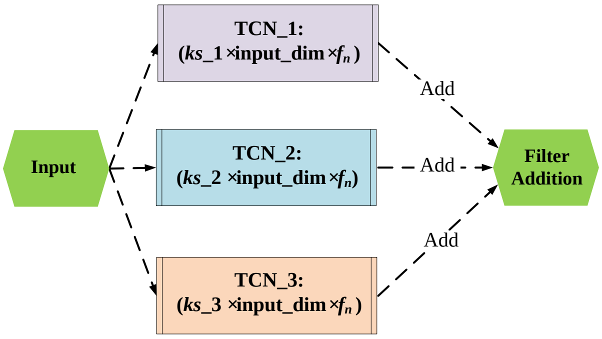

TCN_1:
($ks$_1 ×input_dim ×$f_n$)
TCN_2:
($ks$_2 ×input_dim ×$f_n$)
TCN_3:
($ks$_3 ×input_dim ×$f_n$)
Input
Add
Add
Add
Filter Addition

**Figure 5.** An illustration of the multi-scale TCNs block.

- *2. Transformer block*

TCN modules are adept at capturing the local anomalous fluctuations and spatial relationships hidden in air quality data, but their capacity to model long-term dynamic change characteristics is weak. The limited prediction performance primarily stems from TCN's inability to effectively discriminate the relative importance of each time series and relevant features when processing multi-scale air quality data across varying time steps and representative features. They also cannot adaptively weaken the impact of the noise information, which further limits the model performance.

This paper employs the transformer encoder module to address the limitations inherent in the TCN network. Specifically, it leverages the encoder component from the classical transformer architecture to construct a more lightweight model framework [51,52], which exhibits greater adaptability to the task of long-term feature extraction in air quality prediction scenarios. The transformer encoder architecture incorporates two fundamental components: a multi-head self-attention (MSA) mechanism for capturing global dependencies and a position-wise feed-forward network (FFN) for feature transformation, with the complete network structure illustrated in Figure 6. Notably, each sub-module incorporates residual connections [23] for information propagation, with an additional requirement for layer normalization operations to optimize the learning process. The computation of the multi-head self-attention mechanism is elaborated through the following procedure.

$$Q = xW^q, K = xW^k, V = xW^v \tag{6}$$

In this Equation (6), $x$, $Q$, $K$, and $V$ denote the input sequence, keys vector, values vector, and queries vector, respectively. The weight matrices, i.e., $W^q, W^k \in R^{d_{model} \times d_k}$ and $W^v \in R^{d_{model} \times d_v}$ are learnable parameters, while $d_k$ stands for the dimensionality of the key vector space.

Subsequently, the self-attention (SA) scores are calculated through a scaled dot product operation through the scaled dot product function incorporating a normalization factor of $1/\sqrt{d_k}$, and the corresponding computational procedure is depicted as follows:

$$SA = Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{7}$$

where, $QK^T/\sqrt{d_k}$ denotes the attention weight matrix that computes pairwise relationships between $Q$ and $K$ vectors, subsequently normalized through *Softmax* activation to obtain a probabilistic attention distribution. Through the above complex calculations, the output of a head can be obtained. Nevertheless, relying solely on single-head attention for feature extraction tends to be inadequate. Consequently, integrating multi-subspace feature representations becomes essential for achieving comprehensive information extraction and maintaining model robustness. To address this constraint, the MSA mechanism is implemented through parallel computation of $H$ independent attention heads, whose outputs are subsequently concatenated to form a comprehensive representation. Accordingly, the calculation of MSA can be expressed as follows:

$$\begin{aligned} MSA &= Multi - Head(Q, K, V) \\ &= Concat(head_1, head_2, \ldots, head_i, \ldots, head_H) \end{aligned} \tag{8}$$

$$head_i = Attention(Q, K, V) \tag{9}$$

Upon completion of the computations pertaining to the MSA mechanism, the output corresponding to this sub-block is derived via residual connections and layer normalization, which subsequently serves as the input feature to the feed-forward network block. Ultimately, the extracted features of the transformer block can be obtained, represented as $h_{transformer}$.

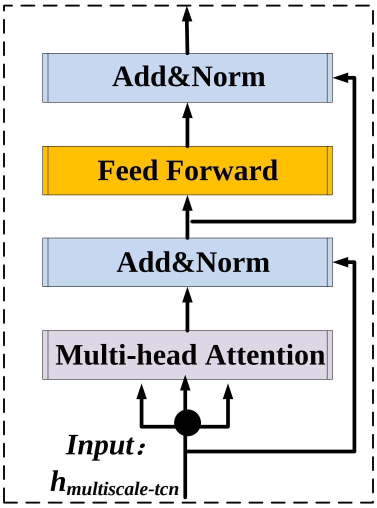


**Figure 6.** The architecture of the transformer.

### 3.3. Step 3. PM Prediction

To derive the final $PM_{2.5}$ concentration, the extracted features from previous layers, denoted as $h_{transformer}$, are input into the prediction module of $PM_{2.5}$ concentrations, which comprises the flatten and linear layers. This step serves to accomplish the inference process that maps the extracted features from previous layers to the $PM_{2.5}$ concentration at the subsequent moment. Such an inference process can be formally expressed by Equations (10) and (11), as follows:

$$h = Flatten\left(h_{transformer}\right) \tag{10}$$

$$PM_{pred} = ReLU(W \cdot h + b) \tag{11}$$

Here, $W$ and $b$ denote the trainable weight matrix and bias vector, respectively, while $PM_{pred}$ represents the output of the predicted $PM_{2.5}$ concentration value. In the training stage, we employ the Mean Squared Error (MSE) as the objective function to quantify the regression performance of the model, as formally defined in Equation (12):

$$Loss_{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(PM_i^{pred} - PM_i^{true}\right)^2 \tag{12}$$

## 4. Experimentation and Results

To verify the prediction accuracy of the proposed model, this section performs comprehensive comparative experiments on publicly available air quality datasets. Specifically, we first present the experimental datasets and relevant evaluation indicators, followed by an introduction to the baseline methods and model hyperparameter configurations employed for comparative analysis. Finally, we elaborate on the experimental results, along with associated ablation experiments and parameter sensitivity analysis.

### 4.1. Datasets Description and Evaluation Indicators

The study utilizes air quality monitoring data collected from four major Chinese metropolitan cities, including Beijing, Shanghai, Guangzhou, and Chengdu [53]. By combing through the relevant research literature, we can find that they either use fewer datasets [54], e.g., only one dataset, or focus on a specific region, such as India [19] or China [55], Europe [56,57], and university campuses [58]. Therefore, it is reasonable to use four sets of data from a specific region, i.e., China, to verify the generalization of the model. The collected dataset comprises temporally-resolved air quality measurements recorded hourly from January 2010 to December 2015, with complete timestamps (i.e., year, month, day, hour, season) and spatial coverage across multiple monitoring regions, including the Nongzhanguan, Dongsihuan, and US-posts sites in Beijing. Since the air pollutants data recorded in US-posts is relatively complete, this paper uses this column as the target column. In addition, since $PM_{2.5}$ concentration is strongly impacted by meteorological conditions, the complementary hourly meteorological data is also obtained and collated from the weather data of airports and Central Meteorological Agency (CMA) site in Guangzhou. Concretely, meteorological data includes Dew Point, Temperature, Humidity, Pressure, etc. Since these datasets have missing data, denoted as NA, we choose to directly delete the consecutive NAs in the early sampling period, and for the remaining NAs, this paper uses easy-to-implement mean interpolation to replace them. It is important to note that there are many time series interpolation methods, such as KNN, GAN, and nearest interpolation [59,60]. In this manuscript, we did not pay too much attention to which interpolation method is more effective in the air quality dataset, but focused on using a traditional mean interpolation method [61,62] to solve the problem of missing values. To address the difficulty of selecting between different interpolation methods, future work could continue to explore and design specialized air quality data interpolation methods to further improve prediction performance. For model evaluation, the dataset was partitioned into training (70%) and testing (30%) datasets, with detailed statistical characteristics summarized in Table 1. In addition, MSTTNet and all baseline methods are conducted in the identical experimental environment, which adopted a desktop equipped with a Core i5-8500 CPU and 8 GB RAM.

**Table 1.** Data description for experimental datasets.

| Variable Type | Variable Name | Data Type |
|---|---|---|
| Air quality data | $PM_{2.5}$ | numerical |
| Meteorological data | Dew Point | numerical |
| | Temperature | numerical |
| | Humidity | numerical |
| | Pressure | numerical |
| | Combined wind direction | Categorical (N/E/S/W/SE/NE/SW/NW) |
| | Cumulated wind speed | numerical |
| | hourly precipitation | numerical |
| | Cumulated precipitation | numerical |
| Timestamp | year | numerical |
| | month | numerical |
| | day | numerical |
| | hour | numerical |
| | season | numerical |

*Evaluation indicators*: The model's prediction performance is quantitatively evaluated using the following four indicators, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and $R^2$. Their corresponding formulae are specified as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i^{pred} - Y_i^{real}\right| \tag{13}$$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|Y_i^{pred} - Y_i^{real}\right| \tag{14}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i^{pred} - Y_i^{real}\right)^2} \tag{15}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(Y_i^{pred} - Y_i^{real}\right)^2}{\sum_{i=1}^{n}\left(\overline{Y}^{real} - Y_i^{real}\right)^2} \tag{16}$$

Among them, $Y_i^{pred}$ represents the predicted results, and $Y_i^{real}$ indicates the corresponding ground truth. For error indicators (i.e., MAE, MAPE, and RMSE), the model can achieve better performance with a lower value. In contrast, for the indicator $R^2$, the model can achieve better performance with a larger value, and the upper limit value is 1.

### *4.2. Baseline Models and Parameter Settings*

A set of models within this field that have demonstrated superior predictive performance in prior research are constructed and subjected to a rigorous comparative analysis against the proposed prediction framework, MSTTNet. The detailed descriptions of these state-of-the-art models are presented as follows:

MLP [44]: This model, known as an artificial neural network (ANN), is capable of modeling non-linear relationships within the air quality dataset with better accuracy. We implement it using a classical design with a three-layer architecture.

1DCNN [5]: The 1DCNN can detect local correlations (e.g., pollutant concentration fluctuations) within air quality time series through one-dimensional convolutional operations. Our implementation employs a single convolutional layer with learnable filters and non-linear activation functions to extract salient local features.

LSTM [10] and GRU [11]: LSTM and GRU networks, developed as advanced variants of classical RNNs, excel at modeling long-term temporal dependencies. In our implementation, each model utilizes a single-layer architecture employing either the LSTM or GRU unit, respectively.

BiLSTM [18]: As a fusion of forward-propagating and backward-propagating LSTM cells [18], this architecture decodes complex temporal patterns in air quality time series through bidirectional context analysis. Our implementation employs a single BiLSTM layer with shared hidden states.

BiGRU-Attention [63]: This hybrid architecture integrates bidirectional GRU processing with attention mechanisms, enabling enhanced modeling of long-term dependencies in air quality datasets while adaptively suppressing irrelevant features. Our implementation comprises a BiGRU layer coupled with an attention layer.

1DCNN-LSTM [47]: This hybrid architecture synergistically models localized fluctuations by using 1DCNN and long-range temporal dependencies via LSTM from air quality and meteorological datasets. Our implementation comprises a single convolutional layer coupled with an LSTM layer.

1DCNN-BiLSTM-Attention [20]: This hybrid architecture extracts spatiotemporal features from air quality data through: (1) localized pollutant pattern capture by using 1DCNN, (2) bidirectional temporal dependency modeling by BiLSTM, and (3) meteorological feature weighting by attention mechanism.

To achieve better prediction performance, the selection of parameters within the constructed model is important. The parameters of the proposed MSTTNet, i.e., epoch, heads, filters, the number of TCN blocks, and the number of transformer blocks, exert a profound influence on the prediction performance. To meticulously determine these parameters, we employ a grid search strategy, conducting a sequential exploration for each parameter. Specifically, across all air quality datasets, we delineate the search ranges as follows: $[50, 100, 150, 200, 300]$ for epoch, $[16, 32, 64, 128]$ for the filter, $[2, 3, 4, 6, 8]$ for the heads, $[1, 2, 3, 4]$ for the TCN blocks, and $[1, 2, 3, 4]$ for the transformer blocks. Through the search process, we identify the parameter combinations that yielded superior prediction performance, which were subsequently utilized to configure the MSTTNet framework. For clarity, all model hyperparameters and their respective configurations are systematically provided in Table 2. For the baseline models, we adhere to the default parameter configurations documented in the relevant references.

**Table 2.** Parameter setting of MSTTNet for each dataset.

| Parameters | Beijing | Shanghai | Chengdu | Guangzhou |
|---|---|---|---|---|
| Epoch | 200 | 150 | 200 | 300 |
| Head | 2 | 2 | 3 | 2 |
| Filter | 32 | 32 | 32 | 32 |
| TCN block | 2 | 2 | 2 | 2 |
| Transformer block | 2 | 2 | 2 | 2 |
| Kernel size | (1, 3, 5) | (1, 3, 5) | (1, 3, 5) | (1, 3, 5) |
| Time lag | 24 | | | |
| Optimizer | Adam | | | |
| Batch size | 32 | | | |
| Learning rate | 0.001 | | | |

### *4.3. Prediction Results Analysis and Comparisons*

This section conducts a systematic comparative analysis of MSTTNet against baseline models through comprehensive experimental evaluations. The key findings are summarized as follows.

We first present the numerical prediction results of each model across the four air quality datasets in Table 3. As is evident from the table, the proposed MSTTNet attains the optimal prediction performance, yielding superior results with lower error indicators compared to the baseline models across most cases (14/16). This phenomenon is primarily ascribed to the robust feature extraction capability of MSTTNet. The proposed multi-scale, multichannel architecture enables the effective extraction of complex features in air quality data through its hierarchical feature learning mechanism, thereby facilitating accurate predictions. Furthermore, predicting air quality proves to be a challenging task for all baseline models, as they fail to achieve higher prediction performance and better fitting effects. Concretely, the baseline models demonstrate significant performance variations when evaluated across distinct air quality datasets, revealing limited generalization capability. For instance, BiLSTM obtains the best performance in 2 out of 16 cases: MAPE value (18.40) for the *Shanghai* dataset and MAPE value (17.10) for the *Chengdu* dataset. As a model with a simple structure relative to CNN-LSTM and 1DCNN-BiLSTM-Attention, BiLSTM achieves the best prediction performance among all baseline models. This phenomenon explains that increased model complexity does not necessarily translate to significant performance gains. BiGRU-Attention achieves better prediction performance in 2 out of 16 cases, i.e., the RMSE (11.40) and $R^2$ (0.9103) values for the *Shanghai* dataset, compared to all baseline methods. In addition, 1DCNN-LSTM achieves better prediction performance in 1 out of 16 cases, i.e., the MAPE value (22.08) for the *Guangzhou* dataset, compared to all baseline methods. Nevertheless, the prediction accuracy of these comparative models in the remaining tasks is deemed unsatisfactory.

Additionally, since MLP, LSTM, and GRU have limited deep feature extraction capabilities, their prediction performance is unsatisfactory, and shows high prediction errors. Although 1DCNN-BiLSTM-Attention can model both local and long-range dynamic features, its basic architecture hinders effective prediction. This limitation likely stems from potential information loss during feature extraction, compromising data integrity. Relative to these baselines, MSTTNet employs the multi-scale architecture operating at different scales to extract local and global representative features hidden within the air quality data. It facilitates more accurate predictions, resulting in the observed performance advantages.

Next, we calculate the average improvement percentages of MSTTNet over baseline models presented in the Table 4, to intuitively demonstrate its prediction capability and generalization across diverse air quality time series. As shown, MSTTNet achieves significant performance gains across all four indicators. Specifically, compared to optimal baseline model BiLSTM, MSTTNet improves MAE, RMSE, MAPE, and $R^2$ by 2.42%, 2.17%, 2.87%, and 0.34%, respectively. Compared to the BiGRU-Attention model, the improvements are 4.17%, 3.36%, 3.57%, and 0.52%, respectively. In addition, compared to the shallow model MLP, MSTTNet improves MAE, RMSE, MAPE, and $R^2$ by 6.36%, 5.70%, 10.62%, and 1.04%, respectively. Notably, comparative analysis reveals that the 1DCNN-BiLSTM-Attention hybrid model demonstrates significantly poor prediction performance compared to both MSTTNet and simple models (i.e., LSTM and GRU). As previously discussed, the model's fundamental architecture demonstrates limited capability in simultaneously extracting both local and global features, resulting in inevitable information loss during feature extraction. In summary, MSTTNet's multi-scale architecture and comprehensive modeling capabilities mitigate information loss, enabling the effective extraction of both local and global dynamic features from air quality datasets.

**Table 3.** Prediction results for each model in four experimental datasets.

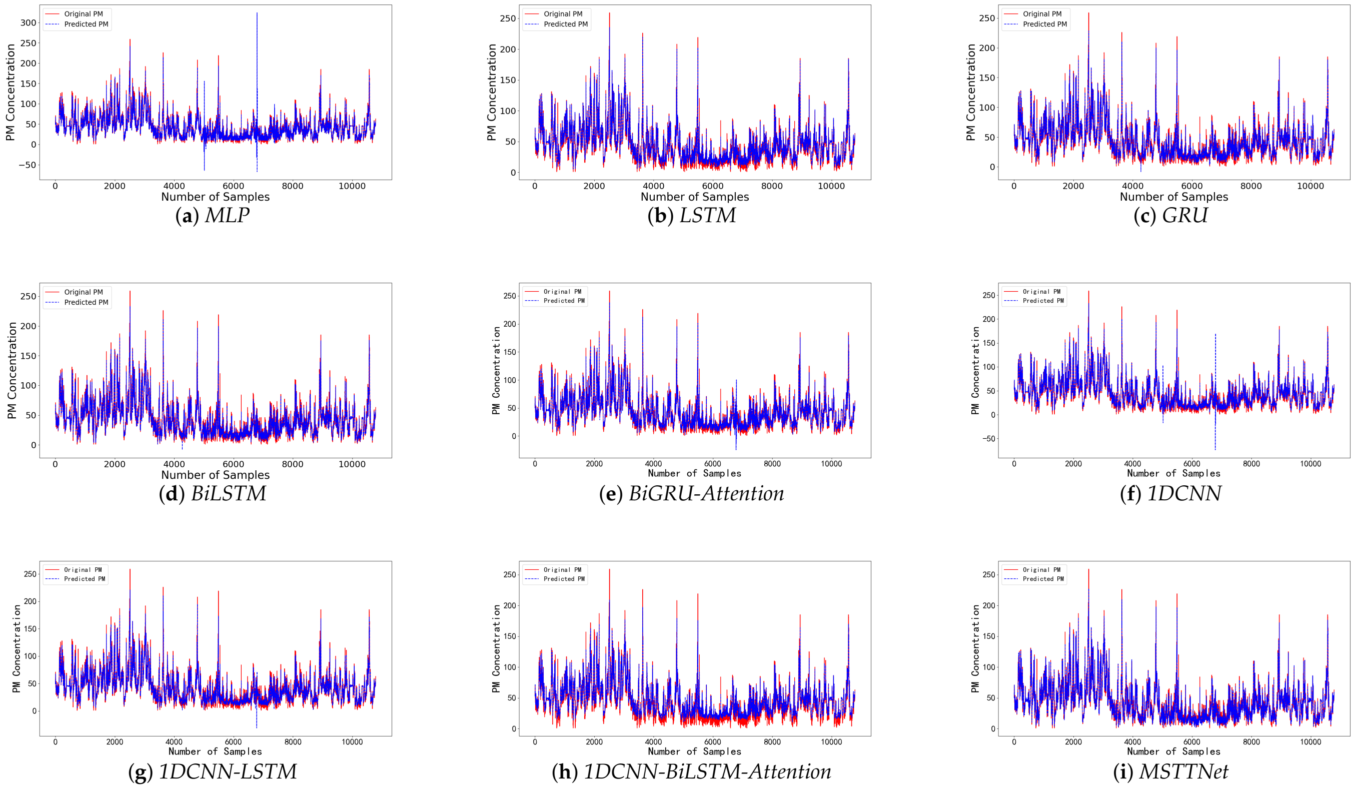
| Datasets | Indicators | MLP | LSTM | GRU | BiLSTM | BiGRU-Attention | 1DCNN | 1DCNN-LSTM | 1DCNN-BiLSTM-Attention | MSTTNet (Proposed) |
|---|---|---|---|---|---|---|---|---|---|---|
| Beijing | MAE | 11.85 | 11.59 | 11.95 | 11.37 | 11.71 | 11.81 | 11.69 | 13.93 | **11.06** |
| | RMSE | 21.79 | 21.61 | 22.13 | 21.37 | 21.85 | 21.39 | 21.53 | 22.68 | **20.85** |
| | MAPE | 29.71 | 26.72 | 27.80 | 25.26 | 25.40 | 29.12 | 26.18 | 35.08 | **23.37** |
| | $R^2$ | 0.9337 | 0.9348 | 0.9316 | 0.9369 | 0.9346 | 0.9361 | 0.9353 | 0.9326 | **0.9393** |
| Shanghai | MAE | 7.21 | 7.04 | 7.10 | 6.98 | 7.07 | 8.63 | 7.58 | 7.70 | **6.82** |
| | RMSE | 11.56 | 11.55 | 11.51 | 11.46 | 11.40 | 12.50 | 11.62 | 11.71 | **11.09** |
| | MAPE | 19.86 | 18.68 | 18.62 | **18.40** | 18.81 | 28.21 | 22.83 | 23.39 | 19.31 |
| | $R^2$ | 0.9079 | 0.9080 | 0.9088 | 0.9095 | 0.9103 | 0.8920 | 0.9064 | 0.9056 | **0.9152** |
| Chengdu | MAE | 9.00 | 8.94 | 8.87 | 8.68 | 8.78 | 9.10 | 8.89 | 9.76 | **8.53** |
| | RMSE | 13.11 | 12.79 | 12.82 | 12.63 | 12.76 | 13.09 | 12.78 | 13.51 | **12.38** |
| | MAPE | 18.62 | 18.50 | 17.42 | **17.10** | 17.13 | 19.43 | 18.13 | 22.34 | 17.63 |
| | $R^2$ | 0.9325 | 0.9356 | 0.9353 | 0.9373 | 0.9360 | 0.9325 | 0.9358 | 0.9283 | **0.9398** |
| Guangzhou | MAE | 6.13 | 5.83 | 5.87 | 5.80 | 5.91 | 6.08 | 5.83 | 6.61 | **5.63** |
| | RMSE | 9.36 | 8.66 | 8.70 | 8.62 | 8.81 | 9.24 | 8.71 | 9.25 | **8.53** |
| | MAPE | 22.99 | 22.30 | 22.46 | 22.72 | 22.68 | 24.21 | 22.08 | 29.70 | **19.98** |
| | $R^2$ | 0.8958 | 0.9107 | 0.9100 | 0.9115 | 0.9077 | 0.8981 | 0.9098 | 0.8998 | **0.9135** |

* The best-performing results across all models are highlighted in bold, while suboptimal results are indicated with underlining.

**Table 4.** Average improvement effects of MSTTNet on four experimental datasets.

| | Indicators | MLP | LSTM | GRU | BiLSTM | BiGRU-Attention | 1DCNN | 1DCNN-LSTM | 1DCNN-BiLSTM-Attention |
|---|---|---|---|---|---|---|---|---|---|
| Imp. | MAE | 6.36% | 3.93% | 4.83% | 2.42% | 4.17% | 10.25% | 5.72% | 14.87% |
| | RMSE | 5.70% | 3.05% | 3.70% | 2.17% | 3.36% | 6.73% | 3.23% | 7.38% |
| | MAPE | 10.62% | 6.06% | 5.51% | 2.87% | 3.57% | 19.50% | 9.60% | 26.15% |
| | $R^2$ | 1.04% | 0.51% | 0.60% | 0.34% | 0.52% | 1.36% | 0.56% | 1.13% |

Ultimately, owing to space constraints, Figure 7 presents a visualization of fitting effects between MSTTNet and baseline models, with the *Guangzhou* dataset serving as a representative case study. As shown, the MSTTNet model consistently and accurately captures future variation trends even when confronted with highly variable air quality data, demonstrating superior fitting capabilities with low deviations. This is particularly evident in its accurate fitting of several anomalous points within Figure 7. Conversely, baseline models frequently deviate significantly when faced with local irregular fluctuations, failing to achieve effective fitting and exhibiting high error. For instance, MLP, 1DCNN, and BiGRU-Attention models show abnormal fitting for some time point data, e.g., in the data sample interval [6000, 8000], which will also lead to an increase in model fitting error. LSTM and 1DCNN-BiLSTM-Attention models also show large errors in fitting at some sample points, e.g., in the sample interval [5000, 7000]. Although BiLSTM shows relatively stable fitting ability at most sample points, it also shows insufficient ability at some sample points, e.g., around sample point 4200. This inability to effectively mitigate the impact of non-stationary variations during feature extraction is the chief culprit of their higher fitting errors. Undue focus on these irregular data points significantly weakens the prediction capability of the models. Furthermore, the selected time slot that contains 200 pieces of sample data predicted by the competitive models and MSTTNet in *Guangzhou* dataset are also depicted in Figure 8. We can observe that air quality prediction is a challenging and complex task for all models. Each model cannot accurately capture the trend of PM concentration. However, for MSTTNet, the deviation between the prediction values and the actual values remains relatively minimum amplitude in most cases, compared with the baseline models. The comprehensive evaluations, incorporating numerical analysis, average improvement effects, and fitting results, demonstrate MSTTNet's statistically significant performance superiority. These findings empirically validate that hierarchical multi-scale local and global feature learning is crucial for achieving accurate air quality predictions.



(**a**) *MLP*
(**b**) *LSTM*
(**c**) *GRU*
(**d**) *BiLSTM*
(**e**) *BiGRU-Attention*
(**f**) *1DCNN*
(**g**) *1DCNN-LSTM*
(**h**) *1DCNN-BiLSTM-Attention*
(**i**) *MSTTNet*

**Figure 7.** Visualization of the fitting effects on the *Guangzhou* dataset for the proposed MSTTNet and comparative models.

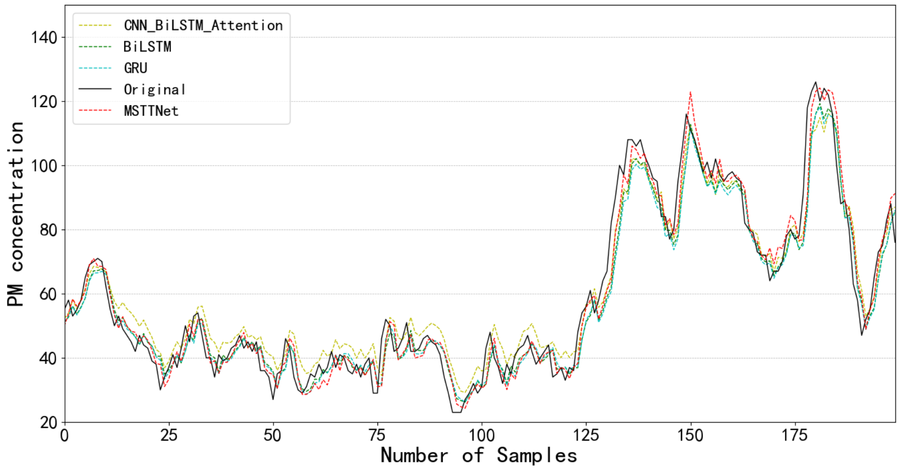
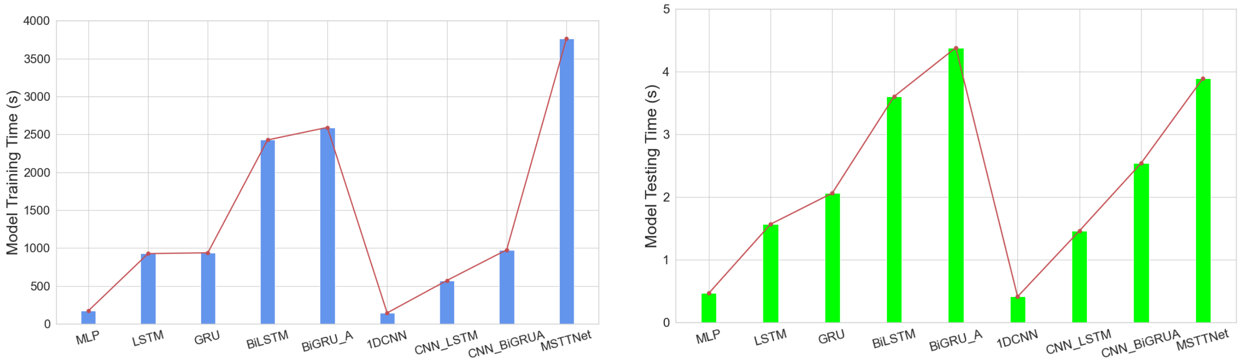


**Figure 8.** The fitting effects of partial subseries for MSTTNet and competitive models.

## 4.4. *Performance Overhead Analysis*

The computational efficiency of the models is evaluated by comparing their training and inference times on the *Shanghai* dataset, as shown in Figure 9. Under identical training conditions, MSTTNet requires the longest training time, about 3762 s, exceeding those of the baseline models, e.g., BiLSTM, GRU, and 1DCNN-BiLSTM-Attention. It should be noted that this training overhead is influenced by factors such as the number of epochs; that is, the larger the epoch, the longer it takes. Nonetheless, during the testing phase, all models demonstrate low inference time. MSTTNet achieves an inference time of 3.89 s, which is comparable to the baseline models. During the test process of these models, BiGRU-Attention takes the longest time, about 4.39 s, while BiLSTM takes 3.6 s to test. However, the performance of these models is still inferior to MSTTNet. Furthermore, the parameter scales of all models are provided in Table 5 on the prediction task of the *Shanghai* dataset, which is exported by the summary() function in the Keras framework. We can observe from Table 5 that MSTTNet does not have large differences in model parameters compared with most baselines. The parameter size of BiLSTM is 375,05, and that of MSTTNet is 43,041, but MSTTNet significantly outperforms BiLSTM. This shows that MSTTNet can achieve superior performance improvements with comparable model complexity.



(**a**) *Training time in the Shanghai dataset* (**b**) *Testing time in the Shanghai dataset*

**Figure 9.** The comparisons of the training and testing time for all models (time is measured in seconds).

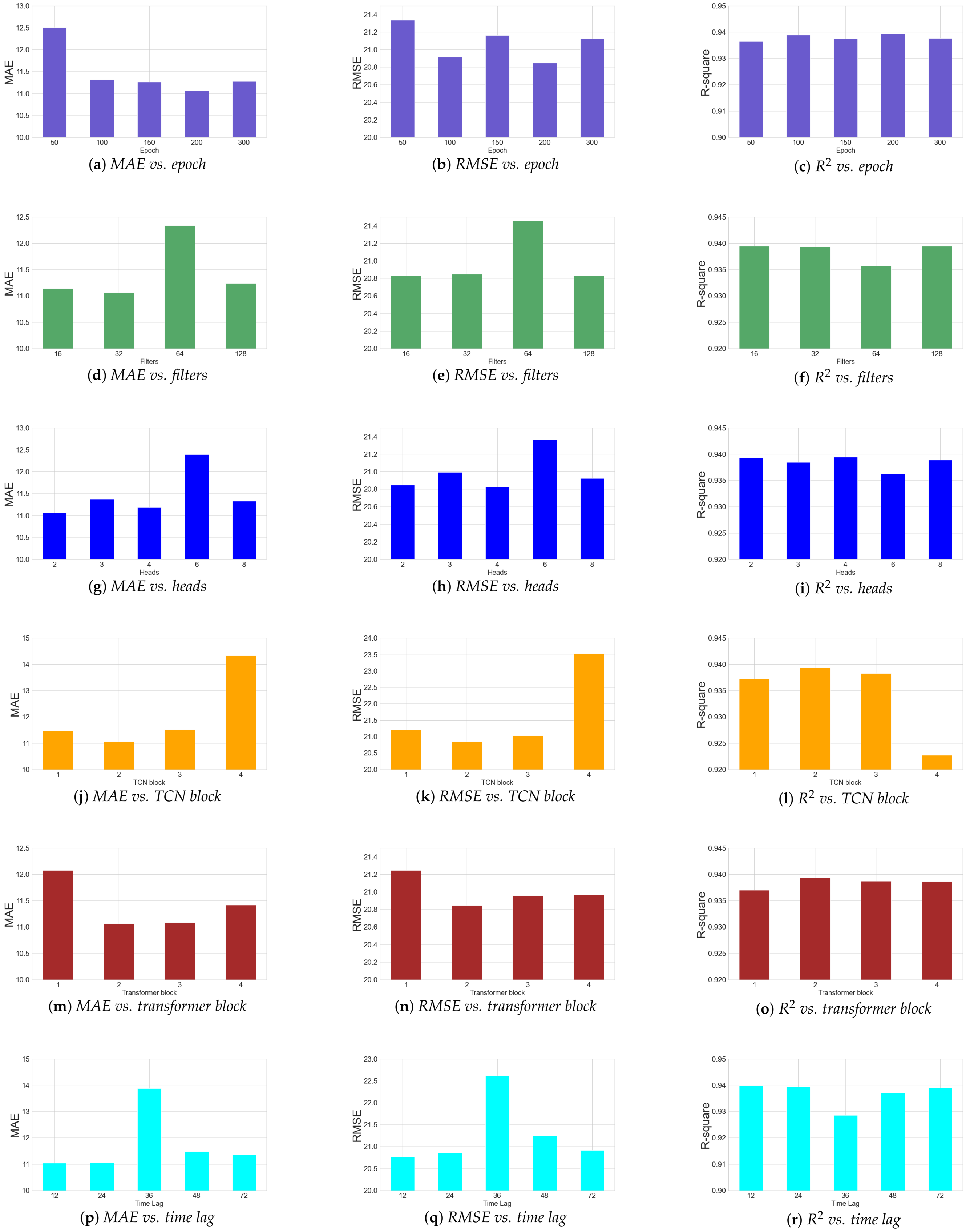
**Table 5.** Parameter scale of each model on the *Shanghai* dataset.

| **Model** | **MLP** | **LSTM** | **GRU** | **BiLSTM** | **BiGRU-Attention** |
|---|---|---|---|---|---|
| Parameter Scale | 1057 | 5281 | 14,273 | 37,505 | 41,049 |
| Model | 1DCNN | 1DCNN-LSTM | 1DCNN-BiLSTM-Attention | MSTTNet | |
| Parameter Scale | 1185 | 9153 | 27,357 | 43,041 | |

### 4.5. Parameter Sensitivity Analysis

This section investigates the impact of parameter selections on model performance to justify the selective model parameters. To this end, the sensitivity analysis is conducted using the *Beijing* dataset as an example. Key parameters, including epoch, filters, heads, the number of transformer blocks, and the number of TCN blocks, are adjusted within reasonable ranges due to their significant influence on performance. The corresponding results for each parameter combination are presented in Figure 10. It should be noted that only the results of MAE, RMSE, and $R^2$ are presented herein, given that the remaining MAPE indicator displays analogous trends and owing to space limitations. The analysis reveals the following key findings:

- Regarding parameter ’epoch’: Simply expanding this parameter does not guarantee improved model accuracy. This performance limitation may derive from either inherent architectural deficiencies of the model or intrinsic data limitations. Beyond a certain point, the model’s performance plateaus and ceases to improve significantly with additional training iterations. For instance, when trained for 300 epochs, the model exhibits degraded performance compared to the 200 epochs configuration, suggesting the onset of overfitting beyond this optimal threshold. Consequently, we set the epoch value to 200 for the *Beijing* dataset to optimize the trade-off between potential accuracy and resource expenditure.
- Regarding parameter ’filter’: The proposed model’s parallel multi-scale design enables robust prediction performance despite employing limited filter quantities, demonstrating efficient feature extraction capability. Nevertheless, escalating filter quantities within the TCN layers fail to yield commensurate performance gains. More critically, this expansion significantly inflates the model’s parameter volume, thereby elevating the risk of overfitting. Furthermore, setting the number of filters excessively high introduces unnecessary computational cost and training time overhead. So, we empirically set the number of filters to 32 for the *Beijing* dataset.
- Regarding parameter ’head’: Increasing the number of heads in the transformer block does not guarantee improved model performance. An inappropriate number of heads, e.g., excessive or insufficient, may induce either overfitting or underfitting, both of which degrade model efficacy. Therefore, we empirically set the number of heads to 2 to balance performance and efficiency.
- Regarding the number of TCN blocks: Changing this structure parameter of the model (e.g., increasing TCN blocks quantities) does not bring desired accuracy improvement. Overly deep and complex model architectures increase the difficulty of model training, and the introduction of gradient issues can easily degrade model performance. More critically, blindly expanding the model structure also inflates the model’s parameter volume, thereby escalating computational costs. In this paper, we configure the TCN block to 2, as this architecture demonstrate acceptable prediction error during hyperparameter tuning.
- Regarding the number of transformer blocks: The transformer block quantities also affect the model prediction performance, as observed in Figure 10m–o. Increasing or decreasing the number of blocks cannot achieve the optimal performance trade-off. Moreover, increasing the number of model blocks will also increase computational overhead without bringing significant performance gains. Finally, we empirically set the number of transformer blocks to 2 to reach a trade-off between model performance and computational complexity.
- Regarding parameter ’time lag’: As the time lag increases, the performance of MST-TNet gradually deteriorates and fluctuates. Increasing the input length of historical data does not necessarily lead to performance improvements. This is mainly because the long historical window contains more noise data, which affects model learning and thus deteriorates model performance. Within the above range, when the time lag is equal to 36 h, the model prediction error is the largest among these parameters. However, the 12 and 24 h time lag windows achieve comparable prediction performance, indicating that the choice of a 24 h time lag in this paper is reasonable.



(**a**) *MAE vs. epoch* (**b**) *RMSE vs. epoch* (**c**) $R^2$ *vs. epoch*

(**d**) *MAE vs. filters* (**e**) *RMSE vs. filters* (**f**) $R^2$ *vs. filters*

(**g**) *MAE vs. heads* (**h**) *RMSE vs. heads* (**i**) $R^2$ *vs. heads*

(**j**) *MAE vs. TCN block* (**k**) *RMSE vs. TCN block* (**l**) $R^2$ *vs. TCN block*

(**m**) *MAE vs. transformer block* (**n**) *RMSE vs. transformer block* (**o**) $R^2$ *vs. transformer block*

(**p**) *MAE vs. time lag* (**q**) *RMSE vs. time lag* (**r**) $R^2$ *vs. time lag*

**Figure 10.** The parameter sensitivity evaluations on the *Beijing* dataset for the proposed MSTTNet architecture.

### *4.6. Ablation Analysis*

This section presents systematic ablation studies to quantitatively evaluate the individual contributions of both the multi-scale TCNs and transformer components within the MSTTNet architecture. The comparative analysis incorporates: (1) the 1DCNN-BiLSTM-Attention baseline, and (2) three architecturally degraded variants of MSTTNet, specifically:

- MSTTNet_1 (single-scale TCNs with transformer): This alternative abandons the multi-scale structure, which only retains the single-channel TCN, but the structure of the transformer block remains unchanged.
- MSTTNet_2 (multi-scale TCNs without transformer): This alternative does not consider the transformer block, which only retains the multi-scale TCNs architecture.
- MSTTNet_3 (transformer without multi-scale TCNs): To verify the effectiveness of the transformer module, this alternative does not consider the multi-scale TCN block, which only retains the transformer block.

In summary, MSTTNet_1 is used to verify the impact of multi-scale modeling capabilities on model prediction, but it still maintains the modeling capabilities of local and global features. MSTTNet_2 is used to verify the importance of global feature modeling capabilities, but it only maintains multi-scale local feature modeling capabilities. MSTTNet_3 is used to verify the importance of local feature modeling capabilities, but it only maintains global feature modeling capabilities. The design of the above ablation version aims to prove that the global–local feature extraction capability is the key to achieving effective prediction.

Table 6 presents the comparative results of four prediction tasks, with significant improvements in each task highlighted in bold. A detailed analysis of the numerical results in Table 6 reveals several important observations. Across the four evaluation indicators, i.e., MAE, RMSE, MAPE, and $R^2$, MSTTNet_1 demonstrates enhanced performance in 3 out of 16 cases, while MSTTNet_2 fails to show any improvements across all 16 cases. MSTTNet_3 achieves better predictions in one case. Nevertheless, the proposed MSTTNet model achieves potential performance enhancements in 12 out of 18 cases, indicating its overall superiority in most prediction tasks.

Among these models, MSTTNet_1 and the proposed MSTTNet models exhibit relatively better prediction capabilities. The above results show that having comprehensive feature extraction capabilities and being able to model both global and local features simultaneously are the keys to achieving effective prediction. The fact that MSTTNet outperforms MSTTNet_2 and MSTTNet_3 emphasizes the criticality of simultaneously modeling local and global relationships in the data during the training process. In contrast, MSTTNet_2 focuses solely on local and spatial features, while MSTTNet_3 concentrates on global features. Local fluctuations are actually difficult to capture accurately, and these data can easily cause the prediction model to “get lost”. Excessive attention to these noisy data will affect the learning of the model and reduce the model’s ability to learn other stable data (except noise data). This result strongly suggests that prediction models to address both types of relationships is essential for achieving accurate predictions. Additionally, fine-grained multi-scale feature extraction capability is also an important component, which can be observed from the comparisons between MSTTNet_1 and the proposed MSTTNet.

Furthermore, when compared with the 1DCNN-BiLSTM-Attention model, all variants have performance advantages, further reflecting the necessity of the designed network architecture. This indicates that the effective feature extraction is crucial. MSTTNet uses multi-scale TCN to replace the traditional 1DCNN, and uses transformer architecture to replace BiLSTM-Attention, which significantly enhances the model’s prediction performance while maintaining computational efficiency.

**Table 6.** Prediction results for each alternative in four experimental datasets.

| Datasets | Indicators | 1DCNN-BiLSTM-Attention | MSTTNet_1 | MSTTNet_2 | MSTTNet_3 | MSTTNet (Proposed) |
|---|---|---|---|---|---|---|
| Beijing | MAE | 13.93 | 11.29 | 11.36 | 12.42 | **11.06** |
| | RMSE | 22.68 | **20.78** | 21.36 | 21.74 | 20.85 |
| | MAPE | 35.08 | 27.10 | 25.19 | 34.73 | **23.37** |
| | $R^2$ | 0.9326 | **0.9397** | 0.9363 | 0.9340 | 0.9393 |
| Shanghai | MAE | 7.70 | 6.94 | 7.63 | 7.28 | **6.82** |
| | RMSE | 11.71 | 11.11 | 11.77 | 11.67 | **11.09** |
| | MAPE | 23.39 | 20.84 | 23.08 | 20.00 | **19.31** |
| | $R^2$ | 0.9056 | 0.9149 | 0.9046 | 0.9062 | **0.9152** |
| Chengdu | MAE | 9.76 | 8.62 | 8.63 | 9.01 | **8.53** |
| | RMSE | 13.51 | 12.42 | 12.58 | 12.99 | **12.38** |
| | MAPE | 22.34 | 18.31 | 16.86 | **16.38** | 17.63 |
| | $R^2$ | 0.9283 | 0.9394 | 0.9378 | 0.9337 | **0.9398** |
| Guangzhou | MAE | 6.61 | 5.85 | 6.08 | 5.66 | **5.63** |
| | RMSE | 9.25 | 8.70 | 8.95 | 8.67 | **8.53** |
| | MAPE | 29.70 | **19.89** | 23.47 | 21.62 | 19.98 |
| | $R^2$ | 0.8998 | 0.9101 | 0.9048 | 0.9105 | **0.9135** |

* The best-performing results across all models are highlighted in bold.

In summary, the empirical results demonstrate that MSTTNet's multi-scale architecture significantly outperforms single-channel schemes. This enhanced performance stems from the model's joint processing of local and global features via dedicated multi-scale TCN and transformer modules, which achieves efficiency and completeness of feature extractions and consequently delivers more stable predictions. We can also empirically observe the complexity of air quality time series modeling. Insufficient or incomplete feature extraction capabilities will greatly affect model performance. Therefore, we recommend that performing PM concentration prediction needs to consider both global and local characteristics, and building a matching network architecture is the primary concern. Additionally, based on these data analytic results, we elaborate on the explicit linkage to practical actions and societal benefits. Accurate PM concentrations prediction can guide the formulation of emission control strategies to combat air pollution problems. For example, implementing congestion charging zones or promoting electric public transport during these specific time windows, e.g., morning rush hours, could be highly effective. Another benefit could be allowing government authorities to issue more accurate and timely health alerts, advising susceptible populations to reduce outdoor activities.

## 5. Conclusions and Future Work

The proposed MSTTNet model overcomes the limitations in traditional time series prediction models (e.g., CNN and LSTM) when capturing the complex structure features of air quality data. MSTTNet architecture excels in incorporating both local and global feature information by adopting multi-scale TCNs and the transformer framework, which contributes to more complete information extraction ability. To validate the architectural design and hyperparameter configurations, we have conducted systematic ablation studies and comprehensive sensitivity analyses, which quantitatively demonstrate the rationality of the model structure and parameter settings. The proposed MSTTNet's performance has been validated on various air quality benchmark datasets in the cities of China, including Beijing, Shanghai, Chengdu, and Guangzhou, by comparing with its eight competition models. Extensive experimental evaluations confirm that the proposed MSTTNet achieves statistically significant accuracy improvements compared to the optimal baseline model.

However, our model may also have some potential limitations. MSTTNet has only been extensively validated on air quality datasets from different regions within China. Different global regions (e.g., Europe or North America) and recent air quality datasets with potentially different emission patterns can be considered, so the generalization of the model is still worth exploring, but MSTTNet will still be applicable for air quality data, including local and global features. Future research may explore more advanced techniques, such as graph learning, adaptive optimization algorithms, and noise filtering techniques, to improve model performance. Moreover, the joint use of satellite remote sensing data and general sensor data for air quality prediction tasks can be further explored in the future, and the use of remote sensing techniques and finer satellite instruments for $PM_{2.5}$ detection/retrieval [64] is also important. Our study provides a valuable ground-based perspective and comparative analysis with their observations. It will complement the spatial coverage of these satellite missions for future studies.

**Author Contributions:** Conceptualization, K.J.; methodology, Z.W.; software, Z.W.; validation, C.Z.; formal analysis, Z.W.; investigation, Z.W.; resources, K.J.; data curation, W.Z.; writing—original draft preparation, Z.W.; writing—review and editing, K.J. and Z.W.; visualization, Z.W.; supervision, K.J.; project administration, K.J. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by Doctoral Research Fund of Nanyang Normal University (Grant No. 2025ZX016) and the Key Scientific Research Projects of Colleges and Universities in Henan Province of China (Grant No. 23A520027 and No. 26A520029).

**Data Availability Statement:** Data generated and acquired for the project will be provided upon request.

**Acknowledgments:** The authors thank the reviewers for their guiding suggestions.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Méndez, M.; Merayo, M.G.; Núñez, M. Machine learning algorithms to forecast air quality: A survey. *Artif. Intell. Rev.* **2023**, *56*, 10031–10066. [CrossRef] [PubMed]
2. Mak, H.W.L.; Ng, D.C.Y. Spatial and socio-classification of traffic pollutant emissions and associated mortality rates in high-density hong kong via improved data analytic approaches. *Int. J. Environ. Res. Public Health* **2021**, *18*, 6532. [CrossRef] [PubMed]
3. Cekim, H.O. Forecasting $PM_{10}$ concentrations using time series models: A case of the most polluted cities in Turkey. *Environ. Sci. Pollut. Res. Int.* **2020**, *27*, 25612–25624. [CrossRef]
4. Sohrab, S.; Csikós, N.; Szilassi, P. Landscape metrics as ecological indicators for $PM_{10}$ prediction in European cities. *Land* **2024**, *13*, 2245. [CrossRef]
5. Wei, Q.; Zhang, H.; Yang, J.; Niu, B.; Xu, Z. $PM_{2.5}$ concentration prediction using a whale optimization algorithm based hybrid deep learning model in Beijing, China. *Environ. Pollut.* **2025**, *371*, 125953. [CrossRef]
6. Du, S.; Li, T.; Yang, Y.; Horng, S.J. Deep air quality forecasting using hybrid deep learning framework. *IEEE Trans. Knowl. Data Eng.* **2019**, *33*, 2412–2424. [CrossRef]
7. Ong, B.T.; Sugiura, K.; Zettsu, K. Dynamically pre-trained deep recurrent neural networks using environmental monitoring data for predicting $PM_{2.5}$. *Neural Comput. Appl.* **2016**, *27*, 1553–1566.
8. Govande, A.; Attada, R.; Shukla, K.K. Predicting $PM_{2.5}$ levels over Indian metropolitan cities using Recurrent Neural Networks. *Earth Sci. Inform.* **2025**, *18*, 1. [CrossRef]
9. Lin, M.D.; Liu, P.Y.; Huang, C.W.; Lin, Y.H. The application of strategy based on LSTM for the short-term prediction of $PM_{2.5}$ in city. *Sci. Total Environ.* **2024**, *906*, 167892. [CrossRef]
10. He, J.; Zhang, S.; Yu, M.; Liang, Q.; Cao, M.; Xu, H.; Liu, Z.; Liu, J. Predicting indoor $PM_{2.5}$ levels in shared office using LSTM method. *J. Build. Eng.* **2025**, *104*, 112407. [CrossRef]

11. Wang, X.; Yan, J.; Wang, X.; Wang, Y. Air quality forecasting using the GRU model based on multiple sensors nodes. *IEEE Sens. Lett.* **2023**, *7*, 6003804. [CrossRef]
12. Liu, B.; Yan, S.; Li, J.; Li, Y.; Lang, J.; Qu, G. A spatiotemporal recurrent neural network for prediction of atmospheric $PM_{2.5}$: A case study of Beijing. *IEEE Trans. Comput. Soc. Syst.* **2021**, *8*, 578–588. [CrossRef]
13. Amnuaylojaroen, T. Prediction of $PM_{2.5}$ in an urban area of northern Thailand using multivariate linear regression model. *Adv. Meteorol.* **2022**, *2022*, 3190484. [CrossRef]
14. Hao, X.; Hu, X.; Liu, T.; Wang, C.; Wang, L. Estimating urban $PM_{2.5}$ concentration: An analysis on the nonlinear effects of explanatory variables based on gradient boosted regression tree. *Urban Clim.* **2022**, *44*, 101172. [CrossRef]
15. Wang, L.; Jin, X.; Huang, Z.; Zhu, H.; Chen, Z.; Liu, Y.; Feng, H. Short-Term $PM_{2.5}$ prediction based on multi-modal meteorological data for consumer-grade meteorological electronic systems. *IEEE Trans. Consum. Electr.* **2024**, *70*, 3464–3474. [CrossRef]
16. Xia, Y.; McCracken, T.; Liu, T.; Chen, P.; Metcalf, A.; Fan, C. Understanding the disparities of $PM_{2.5}$ air pollution in urban areas via deep support vector regression. *Environ. Sci. Technol.* **2024**, *58*, 8404–8416. [CrossRef] [PubMed]
17. Zaman, N.A.F.K.; Kanniah, K.D.; Kaskaoutis, D.G.; Latif, M.T. Improving the quantification of fine particulates ($PM_{2.5}$) concentrations in Malaysia using simplified and computationally efficient models. *J. Clean. Prod.* **2024**, *448*, 141559. [CrossRef]
18. Zhang, M.; Wu, D.; Xue, R. Hourly prediction of $PM_{2.5}$ concentration in Beijing based on Bi-LSTM neural network. *Multimed. Tools Appl.* **2021**, *80*, 24455–24468. [CrossRef]
19. Kumar, S.; Kumar, V. Multi-view Stacked CNN-BiLSTM (MvS CNN-BiLSTM) for urban $PM_{2.5}$ concentration prediction of India's polluted cities. *J. Clean. Prod.* **2024**, *444*, 141259. [CrossRef]
20. Zhu, M.; Xie, J. Investigation of nearby monitoring station for hourly $PM_{2.5}$ forecasting using parallel multi-input 1D-CNN-biLSTM. *Expert Syst. Appl.* **2023**, *211*, 118707. [CrossRef]
21. Wu, S.; Li, H. Prediction of $PM_{2.5}$ concentration in urban agglomeration of China by hybrid network model. *J. Clean. Prod.* **2022**, *374*, 133968. [CrossRef]
22. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271. [CrossRef]
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process Syst.* **2017**, *30*, 6000–6010.
24. Jia, K.; Yu, X.; Zhang, C.; Xie, W.; Zhao, D.; Xiang, J. TTAFPred: Prediction of time to aging failure for software systems based on a two-stream multi-scale features fusion network. *Softw. Qual. J.* **2024**, *32*, 1481–1513. [CrossRef]
25. Liu, X.; Zhi, X.; Zhou, T.; Zhao, L.; Tian, L.; Gao, R.; Luo, J.; Cui, W.; Wang, Q. A holistic air monitoring dataset with complaints and POIs for anomaly detection and interpretability tracing. *Sci. Data* **2025**, *12*, 1288. [CrossRef]
26. Peng, H.; Jiang, B.; Mao, Z.; Liu, S. Local enhancing transformer with temporal convolutional attention mechanism for bearings remaining useful life prediction. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 3522312. [CrossRef]
27. Sun, L.; Liu, M.; Liu, G.; Chen, X.; Yu, X. FD-TGCN: Fast and dynamic temporal graph convolution network for traffic flow prediction. *Inf. Fusion* **2024**, *106*, 102291. [CrossRef]
28. Zhang, Q.; Liu, Q.; Ye, Q. An attention-based temporal convolutional network method for predicting remaining useful life of aero-engine. *Eng. Appl. Artif. Intell.* **2024**, *127*, 107241. [CrossRef]
29. Yin, Z.; Kong, X.; Yin, C. Semi-supervised log anomaly detection based on bidirectional temporal convolution network. *Comput. Secur.* **2024**, *140*, 103808. [CrossRef]
30. Li, L.; Li, Y.; Mao, R.; Li, L.; Hua, W.; Zhang, J. Remaining useful life prediction for lithium-ion batteries with a hybrid model based on TCN-GRU-DNN and dual attention mechanism. *IEEE Trans. Transp. Electrif.* **2023**, *9*, 4726–4740. [CrossRef]
31. Li, Z.; Xie, Y.; Zhang, W.E.; Wang, P.; Zou, L.; Li, F.; Luo, X.; Li, C. Disentangle interest trend and diversity for sequential recommendation. *Inf. Process. Manag.* **2024**, *61*, 103619. [CrossRef]
32. Akbar, S.; Zou, Q.; Raza, A.; Alarfaj, F.K. iAFPs-Mv-BiTCN: Predicting antifungal peptides using self-attention transformer embedding and transform evolutionary based multi-view features with bidirectional temporal convolutional networks. *Artif. Intell. Med.* **2024**, *151*, 102860. [CrossRef]
33. Li, H.; Wu, X.J. CrossFuse: A novel cross attention mechanism based infrared and visible image fusion approach. *Inf. Fusion* **2024**, *103*, 102147. [CrossRef]
34. Guo, Z.; Liu, Q.; Zhang, L.; Li, Z.; Li, G. L-tla: A lightweight driver distraction detection method based on three-level attention mechanisms. *IEEE Trans. Reliab.* **2024**, *73*, 1731–1742. [CrossRef]
35. Kang, H.; Kang, P. Transformer-based multivariate time series anomaly detection using inter-variable attention mechanism. *Knowl.-Based Syst.* **2024**, *290*, 111507. [CrossRef]
36. Sheng, Z.; Cao, Y.; Yang, Y.; Feng, Z.K.; Shi, K.; Huang, T.; Wen, S. Residual temporal convolutional network with dual attention mechanism for multilead-time interpretable runoff forecasting. *IEEE Trans. Neural Netw. Learn Syst.* **2024**, *36*, 8757–8771. [CrossRef]

37. Yuan, X.; Luo, Z.; Zhang, N.; Guo, G.; Wang, L.; Li, C.; Niyato, D. Federated Transfer Learning for Privacy-Preserved Cross-City Traffic Flow Prediction. *IEEE Trans. Intell. Transp. Syst.* **2025**. [CrossRef]
38. Zhang, Z.; Song, W.; Wu, Q.; Sun, W.; Li, Q.; Jia, L. A novel local enhanced channel self-attention based on Transformer for industrial remaining useful life prediction. *Eng. Appl. Artif. Intell.* **2025**, *141*, 109815. [CrossRef]
39. Luo, Q.; He, S.; Han, X.; Wang, Y.; Li, H. LSTTN: A long-short term transformer-based spatiotemporal neural network for traffic flow forecasting. *Knowl.-Based Syst.* **2024**, *293*, 111637. [CrossRef]
40. Model, A. Forecasting Air Quality of Delhi Using. *Advances in Data Sciences, Security and Applications: Proceedings of ICDSSA 2019*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 612, p. 315.
41. Aladağ, E. Forecasting of particulate matter with a hybrid ARIMA model based on wavelet transformation and seasonal adjustment. *Urban Clim.* **2021**, *39*, 100930. [CrossRef]
42. Abdullah, S.; Napi, N.N.L.M.; Ahmed, A.N.; Mansor, W.N.W.; Mansor, A.A.; Ismail, M.; Abdullah, A.M.; Ramly, Z.T.A. Development of multiple linear regression for particulate matter ($PM_{10}$) forecasting during episodic transboundary haze event in Malaysia. *Atmosphere* **2020**, *11*, 289. [CrossRef]
43. Zhou, W.; Wu, X.; Ding, S.; Cheng, Y. Predictive analysis of the air quality indicators in the Yangtze River Delta in China: An application of a novel seasonal grey model. *Sci. Total Environ.* **2020**, *748*, 141428. [CrossRef]
44. Talepour, N.; Birgani, Y.T.; Kelly, F.J.; Jaafarzadeh, N.; Goudarzi, G. Analyzing meteorological factors for forecasting $PM_{10}$ and $PM_{2.5}$ levels: A comparison between MLR and MLP models. *Earth Sci. Inform.* **2024**, *17*, 5603–5623. [CrossRef]
45. Zheng, Y.; Yi, X.; Li, M.; Li, R.; Shan, Z.; Chang, E.; Li, T. Forecasting fine-grained air quality based on big data. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, 10–13 August 2015; pp. 2267–2276.
46. Samal, R.; Krishna, K. Auto imputation enabled deep Temporal Convolutional Network (TCN) model for $PM_{2.5}$ forecasting. *EAI Endorsed Trans. Scalable Inf. Syst.* **2025**, *12*. [CrossRef]
47. Pak, U.; Ma, J.; Ryu, U.; Ryom, K.; Juhyok, U.; Pak, K.; Pak, C. Deep learning-based $PM_{2.5}$ prediction considering the spatiotemporal correlations: A case study of Beijing, China. *Sci. Total Environ.* **2020**, *699*, 133561. [CrossRef] [PubMed]
48. Qi, Y.; Li, Q.; Karimian, H.; Liu, D. A hybrid model for spatiotemporal forecasting of $PM_{2.5}$ based on graph convolutional neural network and long short-term memory. *Sci. Total Environ.* **2019**, *664*, 1–10. [CrossRef]
49. Ren, Y.; Wang, S.; Xia, B. Deep learning coupled model based on TCN-LSTM for particulate matter concentration prediction. *Atmos Pollut. Res.* **2023**, *14*, 101703. [CrossRef]
50. Li, A.; Wang, Y.; Qi, Q.; Li, Y.; Jia, H.; Zhou, X.; Guo, H.; Xie, S.; Liu, J.; Mu, Y. Improved $PM_{2.5}$ prediction with spatio-temporal feature extraction and chemical components: The RCG-attention model. *Sci. Total Environ.* **2024**, *955*, 177183. [CrossRef]
51. Nirmala, G.; Nayudu, P.P.; Kumar, A.R.; Sagar, R. Automatic cervical cancer classification using adaptive vision transformer encoder with CNN for medical application. *Pattern Recogn.* **2025**, *160*, 111201. [CrossRef]
52. Liu, Z.; Feng, Y.; Liu, H.; Tang, R.; Yang, B.; Zhang, D.; Jia, W.; Tan, J. TVC Former: A transformer-based long-term multivariate time series forecasting method using time-variable coupling correlation graph. *Knowl.-Based Syst.* **2025**, *314*, 113147. [CrossRef]
53. Liang, X.; Li, S.; Zhang, S.; Huang, H.; Chen, S.X. $PM_{2.5}$ data reliability, consistency, and air quality assessment in five Chinese cities. *J. Geophys. Res.* **2016**, *121*, 10–220. [CrossRef]
54. Lu, Y.; Wang, J.; Wang, D.; Yoo, C.; Liu, H. Incorporating temporal multi-head self-attention convolutional networks and LightGBM for indoor air quality prediction. *Appl. Soft. Comput.* **2024**, *157*, 111569. [CrossRef]
55. Zou, R.; Huang, H.; Lu, X.; Zeng, F.; Ren, C.; Wang, W.; Zhou, L.; Dai, X. PD-LL-Transformer: An Hourly PM2. 5 Forecasting Method over the Yangtze River Delta Urban Agglomeration, China. *Remote Sens.* **2024**, *16*, 1915. [CrossRef]
56. Sohrab, S.; Csikós, N.; Szilassi, P. Effect of geographical parameters on PM10 pollution in European landscapes: A machine learning algorithm-based analysis. *Environ. Sci. Eur.* **2024**, *36*, 152. [CrossRef]
57. Shetty, S.; Schneider, P.; Stebel, K.; Hamer, P.D.; Kylling, A.; Berntsen, T.K. Estimating surface $NO_2$ concentrations over Europe using Sentinel-5P TROPOMI observations and Machine Learning. *Remote Sens. Environ.* **2024**, *312*, 114321. [CrossRef]
58. Panaite, F.A.; Rus, C.; Leba, M.; Ionica, A.C.; Windisch, M. Enhancing air-quality predictions on university campuses: A machine-learning approach to PM2. 5 forecasting at the University of Petroșani. *Sustainability* **2024**, *16*, 7854. [CrossRef]
59. Owusu-Sekyere, K.; Chen, Y.; Tian, J.; Wang, J.; Dong, Q.; Wang, Z. A comprehensive study of interpolation methods in electrohydrodynamic cone-jet across diverse liquid conductivities. *Phys. Fluids* **2025**, *37*, 082071. [CrossRef]
60. Sun, Y.; Li, J.; Xu, Y.; Zhang, T.; Wang, X. Deep learning versus conventional methods for missing data imputation: A review and comparative study. *Expert Syst. Appl.* **2023**, *227*, 120201. [CrossRef]
61. Xue, Y.; Tang, Y.; Xu, X.; Liang, J.; Neri, F. Multi-objective feature selection with missing data in classification. *IEEE Trans. Emerg. Top. Comput. Intell.* **2021**, *6*, 355–364. [CrossRef]
62. Hung, C.Y.; Wang, C.C.; Lin, S.W.; Jiang, B.C. An empirical comparison of the sales forecasting performance for plastic tray manufacturing using missing data. *Sustainability* **2022**, *14*, 2382. [CrossRef]

63. Chen, Y.; Ye, C.; Wang, W.; Yang, P. Research on air quality prediction model based on bidirectional gated recurrent unit and attention mechanism. In Proceedings of the 4th International Conference on Advances in Image Processing, Chengdu, China, 13–15 November 2020; pp. 172–177.
64. Mak, H.W.L.; Laughner, J.L.; Fung, J.C.H.; Zhu, Q.; Cohen, R.C. Improved satellite retrieval of tropospheric $NO_2$ column density via updating of air mass factor (AMF): Case study of Southern China. *Remote Sens.* **2018**, *10*, 1789. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.